Automatic Tracking Initialization from TriDAR data for Autonomous Rendezvous & Docking

S. Ruela\textsuperscript{a}, D. Ouellet\textsuperscript{b}, T. Luu\textsuperscript{a} and D. Laurendeau\textsuperscript{b}

\textsuperscript{a}Neptec Design Group, Ontario, Canada. \{sruel, tluu\}@neptec.com
\textsuperscript{b}Laval University, Quebec, Canada. \{douellet, lauren\}@gel.ulaval.ca

Abstract

Neptec has developed a vision system for autonomous on-orbit rendezvous and docking that operates from 3D data and does not require cooperative targets. The system combines a TriDAR active 3D sensor and a model based tracking algorithm to calculate relative pose information (6 DOF) in real-time. In collaboration with Laval University and the Canadian Space Agency (CSA), techniques to localize an object in space from 3D data have been developed. These algorithms are necessary to automatically initiate the tracking system and recover if tracking lock is lost. The first approach developed, called Polygonal Aspect Hashing, was designed to operate directly from a sparse disorganized 3D point cloud. This takes advantage of the random access nature of the TriDAR sensor that can acquire data using localized fast scanning waveforms. The second technique developed, uses a geometric hashing feature matching approach.

1. Introduction

Vision systems for on-orbit rendezvous and docking traditionally rely on cooperative markers positioned on the target spacecraft [1],[2]. For some mission concepts, this is not always practical or even possible [3]. The target spacecraft could already be in space without the necessary cooperative targets or could be tumbling or positioned in such a way that the targets are not visible. Research and development in 3D sensing technologies and computer vision at Neptec has led to a vision system that can be used reliably on-orbit for relative navigation without requiring cooperative targets [4],[5]. The system combines a TriDAR (triangulation + LIDAR) 3D active sensor with Neptec’s 3DLASSO model based tracking software. The system requires only knowledge about the target geometry and 3-dimensional data acquired from the sensor to compute the 6 Degree Of Freedom (6DOF) relative pose directly. In order to initialize the tracking software or recover if tracking is lost, a separate process is required. This process is responsible for localizing the target with no a-priori knowledge about its position or orientation in space.

This paper presents novel object localization techniques developed concurrently by Neptec and Laval University’s Computer Vision and Systems Laboratory (CVSL) to perform the initialization of the tracking system. This research is being performed under the Canadian Space Agency (CSA) Space Technologies Development Program (STDP). As opposed to traditional solutions to the general object localization problem, this research focuses on providing a specific solution to on-orbit localization while considering the associated operational constraints.

Section 2 presents an overview of Neptec’s relative navigation sensor solution. Section 3 lists the research objective and the general approach to solving the localization problem. Section 4 presents a point based approach called Polygonal Aspect Hashing. Section 5 details a feature based technique that uses geometric hashing to localize a target object in space.

2. TriDAR System Overview

Neptec’s relative navigation vision system combines a TriDAR active 3D sensor with the 3DLASSO model based tracking software [4],[5],[6]. The system provides 6 Degree of Freedom (DOF) relative pose information in real-time from 3D sensor data. Real-time performance has been achieved by implementing a smart scanning strategy referred to as More Information, Less Data (MILD) where only the necessary data to perform the pose estimation is acquired by the sensor.
2.1. TriDAR Sensor

The TriDAR sensor combines auto-synchronous triangulation and Time-of-Flight (ToF) active ranging techniques within the same optical path. This patented configuration takes advantage of the complementary nature of these two imaging technologies, and allows the system to provide fast and accurate 3-dimensional data at both short and long range [6]. By merging the subsystem’s optical paths, the TriDAR provides the capabilities of two 3D scanners into one compact design. The dual 3D imaging subsystems also share the same control and processing electronics, thus providing further savings compared to using two separate sensors. Furthermore, the TriDAR design is largely based on the Laser Camera System (LCS) active triangulation scanner used to perform inspection of the Space Shuttle’s Thermal Protection System (TPS) [7]. Most of the system is therefore already space qualified. By combining high accuracy short range 3D imaging with the long range capability of LIDAR, TriDAR is a sensor well suited to numerous space applications such as: rendezvous & docking, planetary navigation and landing, site inspection, material classification and vehicle inspection. A TriDAR sensor could be used to perform several of these operations within a single mission, thus providing significant hardware savings.

For rendezvous and docking applications, the TriDAR presents advantages compared to other types of sensors. Primarily, the TriDAR can provide the data accuracy and speed necessary for the critical short range docking phase, while still providing the long range capability required for the approach phase. Secondly, since the LIDAR imaging subsystem of the TriDAR can be used over the entire operational range; it improves reliability by providing redundant range measurements to augment the triangulation data used during final approach. A novel feature of the triangulation optical configuration of the TriDAR is that it provides natural signal attenuation as range decreases. This effectively increases the dynamic range of the sensor and prevents saturation of the detection electronics at short range. Finally, the TriDAR is a random access scanner, meaning that it can directly scan only the area of interest in its field of view. This avoids wasted time acquiring data that will not be used; a key element of the MILD strategy that will be detailed in the following section.

2.2. 3DLASSO Tracking Software

The tracking algorithm referred to as 3DLASSO is compiled in a fully portable, sensor independent model based tracking software library [4]. The algorithm uses a 3D model of the target along with a sparse point cloud from a 3D sensor to compute the 6 Degree Of Freedom (DOF) relative pose in real-time. No cooperative targets such as retro-reflectors are required on the target object. The shape of the target is typically obtained from CAD models but could also be generated from sensor data directly [3], [8], [9].

The key to obtaining real-time performance is the More Information Less Data (MILD) strategy. Using this approach, only the data necessary to obtain the target pose is acquired by the sensor. This strategy maximizes the geometric information obtained while minimizing processing cost (acquisition time, data bandwidth, required memory and data computation).

The MILD paradigm comes from previous research at Neptec which demonstrated that only a few 3D points are necessary to not only calculate the pose of an object, but also to recognize it from a set of potential candidates [8],[9]. This follows the logic that differentiating objects and/or estimating their poses is mainly done by using their largest discriminating geometric features. Therefore, high-resolution 3D data contains a lot of redundant data unnecessary to accomplish the desired high level task. Significant savings can thus be made if only the necessary data is acquired in the first place.

To perform the pose estimation, the data obtained is aligned to the reference model of the target(s) using an innovative approach to the Iterative Closest Point (ICP) algorithm developed at Neptec. This version of the ICP was tailored for real-time operations from unorganized sparse 3D data while still providing good pose accuracy. It was shown that pose accuracy under 1cm / 1degree at a 5Hz update rate can be achieved on a flight processor using this technique [4],[5].

3. Target Localization from 3D data

Since the 3DLASSO tracking system uses an ICP algorithm to calculate the relative pose, the tracking system requires an initial pose. Once in steady state, the system uses the last known position or a prediction from an estimated trajectory as its initial guess. A separate process is therefore required to initialize the tracking system or recover if tracking is lost. Present research is focused on developing a technique for object localization that addresses the specific on-orbit localization problem and associated constraints. The algorithm developed needs to estimate the 6 DOF relative pose of a known target from 3D data without
any initial guess. Since ICP will be used to refine the pose, only a rough estimate is needed from the localization process. In order to be applicable to on-orbit operations, the technique must overcome the following challenges:

- **Processing platform available.** The algorithm must be capable of operating fast enough on a flight computer to enable real-time operations. This implies that localization must be performed in less than a few seconds. Available platform memory must also be considered.
- **Data acquisition and transfer time.** Acquiring 3D data and processing it requires a significant amount of time. It is therefore important that the technique be designed to minimize this factor.
- **Active sensor noise characteristics.** The technique must be robust to the noise and distortion characteristics typical of 3D active sensors.
- **Occlusions.** As the operating range in rendezvous and docking is quite large, it is important that the system be able to operate when the target is only partially visible. This typically occurs in the final docking stage when the target covers more than the sensor field of view or if the target is very large (e.g. Space Station).

### 3.1. Design Approach

In order to provide a real-time solution to the object localization problem, the developed algorithmic approaches diverge from traditional techniques. This is achieved by specifically addressing the on-orbit localization problem as opposed to solving the general problem. This strategy enables the following simplifying assumptions:

1. No segmentation is required since the object is in empty space. I.e. no background clutter is expected.
2. Only one instance of the target object is visible in the scene.

Typical object localization approaches attempt to find one or several instances of a known object in data from a cluttered scene. In the general case, it makes sense to fit the model to the scene because background clutter prohibits the opposite logic. In contrast, under the defined assumptions, it is possible to perform the search in the opposite direction, by best fitting the scene data to the reference model. By reversing the search logic we can simplify and accelerate the entire process.

Object localization research investigated two fundamentally different approaches to solving the problem: point based and feature based. A point based approach finds a pose that best fits the dataset to the model, while a feature based technique matches features extracted from the data to model features. Neptec has developed a point based technique called Polygonal Aspect Hashing. The Computer Vision and Systems Lab (CVSL) at Laval University developed a feature based localization algorithm relying on geometric hashing to find the relative pose. This research was performed concurrently to widen the range of potential approaches tested while reducing overall project risk.

### 4. Polygonal Aspect Hashing

The point based technique developed by Neptec for object localization is called Polygonal Aspect Hashing. A point based technique works directly with the raw dataset and completely bypasses computationally expensive filtering and feature extraction steps. Furthermore, the source data does not need to be arranged in a grid since no convolution operators will be used to extract surface properties or filter the data. Following the same MILD approach as used for the tracking algorithm, the localization system was designed to require only a sparse unorganized 3D data set. This strategy makes maximum use of the large amount of information contained in a small number of 3D measurements. It also minimizes the time required to acquire, transfer and pre-process the data. To obtain the data, the TriDAR sensor is commanded to acquire only a small number of 3D datapoints (100’s) from the area of interest using a dynamically efficient scan pattern such as a Lissajous, Rosette or Spiral.

The Polygonal Aspect Hashing follows a process similar to adding a piece in a jigsaw puzzle. Basically, the input data is aligned to a reference model in various poses that present, at least, partial alignment until a best fit is obtained. The technique requires an offline processing step where a reference database is generated from a 3D model of the target object. The run-time portion of the algorithm then uses that reference database to efficiently localize the target object in space relative to the sensor.

#### 4.1. Search Space Reduction

The pose of the target is an element of a 6 dimension space (3 translations and 3 rotations). The first step to finding the true pose is to reduce the search space to the most likely candidates. This is performed
by keeping only the set of poses that have at least some overlapping surface between the input scan and the reference model (Figure 1).

Polygonal aspect hashing is used to perform the search space reduction step. First, an N point polygon is selected from the 3D points acquired by the sensor. N can be adjusted depending on the object’s geometry, sensor characteristics, desired processing time and outlier tolerance. Typically, 4 to 6 point polygons work well. The points are selected to provide a polygon with a large surface area. This will provide greater pose determination momentum and fewer polygon matches. Once a polygon is selected, the set of possible poses is reduced. Up to a specified tolerance, only poses that line up polygon scan points with corresponding model surface points are considered (Figure 2). Matching polygons are found efficiently using a hash table contained in the reference database generated offline. The key to optimizing performance and minimizing memory use is to keep the reference model information sparse. Only the largest discriminating geometric features need to be stored. This will keep the number of matches small while providing a sufficiently accurate pose estimate for an ICP to perform the fine alignment.

4.2. Testing the pose candidates

For each polygon match, a set of N matched points is obtained. For each matched polygon pairs, the corresponding relative pose (translation and rotation) can be computed. Assuming the points in each matched polygon represent the same physical points on the object surface, the two sets of matched points can be related by a homogenous transformation. Once the set of pose candidates has been generated, the algorithm tests them to assess how well the input 3D points line up with the reference model surface (Figure 3). This surface alignment check has O(n) complexity where n is the number of points in the input point cloud. The pose candidate that aligns the point cloud best with the surface is chosen as the solution. It is also possible to increase the speed of the fit check by first performing a pre-check step with a decimated dataset to further reduce the search space.

After the best fitting pose candidate has been determined, it is possible to repeat the process with more polygons from the same input point cloud or from newly acquired point clouds over time. Using either technique will increase the probability of success of the algorithm as well as increase robustness to outliers. The approach used can be selected depending on operational considerations such as relative motion rates and processing time / power available.
4.3. Test Results

The algorithm was tested using simulated and real sensor data of a half scale model of an International Space Station (ISS) Pressurized Mating Adapter (PMA) connected to a lateral port of an ISS Node (Figure 4). This target was selected for the initial testing because of its realistic geometry and because a physical model was readily available for lab testing. The algorithm offline process was executed on the target object using a 100mm resolution. The offline process generated a 1.5MB polygonal aspect hashing search database file for this model.

Simulated Data

Simulated data was used to initially characterize the algorithm performance. Simulated data can produce very good statistics regarding algorithm performance since many trials with known true pose can be obtained quickly. A common problem with simulated data is the poor realism of the sensor noise model. Active sensors have biased noise and present distortions that are dependent on not only the sensor, but also on the target being scanned. These effects must be taken into account when generating simulated data for vision system algorithm characterization. Neptec’s sensor simulation framework uses a realistic active scanning sensor imaging model that simulates the various parameters affecting the imaging process. Factors such as range noise, laser source divergence, pointing resolution, as well as the sensor’s interactions with target surface shape and material properties are simulated.

The algorithm was tested using 220 different poses of the target object. At each pose, 5 different simulated 3D point clouds were generated from the target and tested independently for a total of 1100 localization tests per sequence. Each simulated 3D point cloud contained less than 900pts and was arranged in a simple 5x4 Lissajous pattern. Note that for a scanning sensor like the TriDAR, acquiring such a point cloud would require less than 100ms, fast enough for real-time operations. The simulated sensor was configured to have noise characteristics that are representative of a typical 3D active sensor. Table 1 summarizes the success rate and processing time of the algorithm as a function of the number of polygons used during the search process. This test was performed on a Pentium M 1.6GHz laptop. Since the processing time of the algorithm varies greatly depending on the number of polygon matches found, the 3σ processing time is presented. As expected, the success rate of the algorithm increases with the number of polygons at the expense of processing time. Table 1 also presents the mean pose error obtained from the algorithm for successful localizations.

<table>
<thead>
<tr>
<th># Polys</th>
<th>Success Rate (%)</th>
<th>Processing Time (3σ)</th>
<th>Mean Pose Error</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>X (mm)</td>
<td>Y (mm)</td>
</tr>
<tr>
<td>1</td>
<td>66</td>
<td>71</td>
<td>55</td>
</tr>
<tr>
<td>2</td>
<td>92</td>
<td>131</td>
<td>47.5</td>
</tr>
<tr>
<td>5</td>
<td>99</td>
<td>352</td>
<td>39.5</td>
</tr>
</tbody>
</table>

Lab Testing

The algorithm was tested in Neptec’s vision lab using a TriDAR sensor and a full size mockup of the target object (Figure 4). The TriDAR was positioned at roughly 20m (maximum range of the facility), 10m, and 4m away from the target object. Note that, at 4m, less than half of the target object is visible. The distances selected are representative of the final stage of a docking mission where the target would go from fully visible (long range) to only partially visible (short range). At each range, 10 different relative poses were tested. For each viewpoint, 3 different 1024 point scans were acquired using different scan patterns (Lissajous, Rosette and Spiral). The scan extents were selected to avoid lab clutter as this would have violated the operational assumptions defined for this system.
As observed in simulations, the success rate of the algorithm increases with the number of polygons tested (Table 2). As expected, the processing time also scales with the number of polygons. Note that the number of points on target remained fairly constant (900-1000) throughout all these tests. Table 2 also shows the absolute mean pose error observed during lab testing for successful localizations. The pose accuracy does not significantly improve when using more polygons, but remains well within the desired range. The mean pose error observed is also consistent with simulation results.

Table 2 Polygonal Aspect Hashing Lab Results

<table>
<thead>
<tr>
<th># Polys</th>
<th>Success Rate (%)</th>
<th>Processing Time (ms)</th>
<th>X (mm)</th>
<th>Y (mm)</th>
<th>Z (mm)</th>
<th>RotX (deg)</th>
<th>RotY (deg)</th>
<th>RotZ (deg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>91</td>
<td>187</td>
<td>50.3</td>
<td>41.8</td>
<td>30.5</td>
<td>1.6</td>
<td>1.4</td>
<td>2.8</td>
</tr>
<tr>
<td>2</td>
<td>93</td>
<td>293</td>
<td>41.3</td>
<td>45.7</td>
<td>28.9</td>
<td>1.52</td>
<td>1.4</td>
<td>2.8</td>
</tr>
<tr>
<td>3</td>
<td>100</td>
<td>773</td>
<td>40.8</td>
<td>47.2</td>
<td>30.3</td>
<td>1.4</td>
<td>1.3</td>
<td>3.1</td>
</tr>
</tbody>
</table>

5. Feature Based Technique

The approach developed by Laval University’s Computer Vision and Systems Laboratory (CSVL) consists firstly in extracting some surface features from the scene. Surface portions corresponding to planes, cylinders and spheres are parameterized (Table 3). Secondly, the pose is estimated by comparing the extracted features against the known model with a Geometric Hashing (GH) algorithm. This algorithm has the reputation to be fast and reliable but requests significant memory space. It is well-known that a feature-based approach requires more points (tens of thousands) than a point-based approach (few hundreds), but the parameterized surfaces offer a very high compression factor, which is desirable when applied to large and complex 3D models.

Table 3 Extracted features and parameters

<table>
<thead>
<tr>
<th>Feature</th>
<th>Parameters</th>
</tr>
</thead>
</table>
| Plane   | Ax + By + Cz + D = 0  
N = (A, B, C), with A² + B² + C² = 1  
D = distance to origin |
| Sphere  | P_center = (X, Y, Z)  
R = radius |
| Cylinder| V_axis = (A, B, C)  
P_axis = (x, y, z)  
R = radius |

5.1. Feature extraction

A part of the challenge of the feature extraction step is to efficiently exploit the data produced by the TriDAR sensor. The TriDAR scans points in a sequential process along the path of a high-order Lissajous pattern which provides a mesh of about 10000 points. Such an approach yields a relatively dense set of points along the scan path and avoids the use of a raster scan pattern which would be 10 times slower than the Lissajous pattern for similar point spacing.

Following the range image acquisition (Figure 5a), the raw data points are filtered; then object contours and edges are extracted (Figure 5b) [11]. Finally, the surface tangents are computed. These tangents are determined by direct least-square fitting of 2D line segments [12] along the curvilinear scan path. The local surface orientations are obtained by merging the tangents at the scan intersections. These operations typically reduce the volume of data by a factor 10 to 20. Since most of them are repeatedly performed on a small domain along the scan path, several time-saving optimizations can be implemented.

The next step is to segment the preprocessed range image in order to find regions with similar geometric properties (Figure 5c). Step discontinuities are exploited to limit regions and look for flat surfaces and for surfaces with constant curvature. The content of each region must be validated and only the sufficiently large regions of interest must be kept to confirm that a planar surface, a cylinder, a sphere or some other geometric shape has been found. Too small regions as well as more complex and hardly parameterized regions are simply discarded from the features. In fact, the whole surface does not need to be fully parameterized but only some features for the pose information retrieval.

![Figure 5 Test object](image)

Each feature provides only partial pose information: the plane provides full orientation and a partial position, but this very common feature is harder to
match than more complex geometric shapes; the sphere defines a fixed location, but does not provide any clue on orientation; and finally, the cylinder is more precise and particularly useful, since it defines a central axis (orientation) perpendicular to its visible surface. Both cylinders and spheres are also described by the value of their radius which helps in the matching process. In addition, cylinders and spheres are good pose invariant features because of their symmetry; while planes have rather limited view angles (the scans must be almost perpendicular to the surface).

5.2. Geometric Hashing

For pose determination, at least three features of the scene must be matched with the model. This match can be achieved by the Geometric Hashing (GH) technique described by [10], [13], [14]. GH is divided into two main steps: the first step is an off-line model processing stage where multiple representations of the object are processed and stored in a Hash-Table (HT); while the second step is the fast online feature matching with the previously computed HT.

The basic idea behind the GH is to be able to represent specific object properties with unique signatures (hashes) which will be triggered only when similar properties appear in the scene. During the intensive off-line processing, all possible signatures from the model must be compiled and stored in the HT. The feature matching step can be performed relatively quickly because only one set of signatures is generated from the scene. Matching results from a voting procedure using the HT content. Weak or null matches can also be detected thus preventing erroneous pose estimates to be generated. A weak match may occur for two reasons: when the object is not present in the scene or when some scene features (mobile features like solar arrays, antennas and robot arms) are not present in the model. To circumvent the latter situation, the voting process should be performed again with different features until a match is found.

The GH approach relies on pose invariant data representation of the surface features. Such data can then be quantized and compared to the content of a pre-computed HT containing the results of the model’s multiple feature combinations. It has already been done with 3D points or surface orientations, but it has never been done with heterogeneous primitives like planes, cylinders and spheres. A common representation using eight parameters is required: a Type, a constant (K) to hold the radius or distance, a normalized orientation (A, B, C) and a 3D point (X,Y,Z). Once quantized, the parameters A, B, C and K define the addresses for the HT, while the coordinates X, Y, Z are stored into an extension list accompanying the HT for a linear search.

The GH bases are unique transforms (rotation and translation) required to change the other features into pose independent parameters. In order to insure the uniqueness of the basis, the computation must involve information from the three selected features: the rotation matrix is defined by the orientations of the first two features; while the third feature is intersected with the other features to define a translation. During the basis definition process, several non compatible or non supported features combinations are eliminated. The valid combinations are typically below 10% of the possibilities (a set of M features will produce up to M!/(M-3)! bases). Depending on the model complexity, it is known that specific features like cylinders and spheres are less frequent than the planar surfaces. We can benefit of this observation by requiring that at least one cylinder (or one sphere) should be present in the basis. It however requires that at least one of these features be visible and detected in the scene. The basis elimination process can improve the HT sparseness and strengthen the pose accuracy.

5.3. Preliminary results

We performed a preliminary test of GH with the Test Object model (Figure 4). This model contains 23 features, including 15 planes and 8 cylinders (extracted with CAD software). This model produces 418 valid bases, more precisely 8% of the 5250 possibilities, with a minimum 60 degree separation of the features forming a basis. The bases transforms use 38KB of storage, but these redundant data can still be easily reduced to a compact 7KB. The HT (25600 cells and 8360 extensions) requires only 172KB. Surprisingly, the complete model for the Test Object occupies a very small 210KB footprint.

6. Conclusion

Neptec’s vision system for Autonomous Rendezvous and Docking (AR&D) combines a TriDAR 3D active sensor and model based tracking algorithms. The vision system uses sparse 3D data from the TriDAR sensor to compute full 6 degree of freedom relative pose information in real-time without requiring cooperative markers on the target spacecraft (e.g. retro-reflectors).

Under the Canadian Space Agency’s Space Technologies Development Program (STDP), algorithms for object localization were developed.
Such algorithms are required to initialize the model based tracking software and recover if tracking is lost. This research was performed concurrently by Neptec and Laval University. Neptec focused on point based solutions while Laval University’s CVSL investigated the use of feature based techniques. This concurrent approach to the research was used to widen and strengthen actual understanding of the problem and to suggest different means of providing a robust solution.

This research focused on solving the specifics of the on-orbit object localization problem as opposed to more traditional research on the general localization problem. By considering a specific scenario (on-orbit localization), simplifying assumptions can be used and specific constraints can be taken into account. Both techniques developed demonstrated great potential for use in Neptec’s AR&D system. These fundamentally different approaches provide more flexibility to deal with mission specific problems such as target geometry, materials and/or sensor characteristics. It is also possible to combine these techniques into a hybrid approach. For example, the feature based technique could be used in the first phase of the point based technique to enhance the set of pose candidates.

7. Acknowledgments

The authors would like to thank the Canadian Space Agency (CSA) for funding and supporting this research.

8. References